

論文2003-40SC-5-5

## 다중 DSP 보드를 이용한 프로그램 가능한 도플러 처리기 (A Programmable Doppler Processor Using a Multiple-DSP Board)

申鉉翼\*, 金煥宇\*\*

(Hyun-Ik Shin and Whan-Woo Kim)

### 요약

도플러 처리(Doppler processing) 기능은 잔류(residue) 클러터(clutter)의 제거뿐만 아니라 위상정합누적(coherent integration)을 수행하므로, 펄스 도플러 레이더에 있어서 가장 핵심적인 역할을 수행한다. 디지털 신호처리기(DSP: digital signal processor)의 성능향상과 더불어 DSP를 이용한 구현이 점점 일반화되어가고 있다. 도플러 처리기가 입력신호를 실시간으로 처리하기 위해서는, 다중 DSP를 이용한 병렬처리 개념이 일반적으로 사용되어야 한다. 본 논문에서는 아날로그 디바이스사의 ADSP21060 8개를 탑재한 Morocco-2 보드를 사용하여 MTI(moving target indicator)필터, 도플러 필터뱅크(DFB: Doppler filter bank) 및 제곱검출기(square-law detector) 등으로 구성된 프로그램 가능한 구조의 도플러 처리기를 구성하였다. 위상정합처리구간(CPI: coherent processing interval) 동안 수신된 입력데이터의 분배(distribution)시간, 출력데이터의 전송(transfer)시간 및 알고리즘 수행에 소요되는 연산시간 등을 수식으로 표현하여, 전체 처리시간과 도플러 처리기 구현에 소요되는 DSP의 수를 예측하였다. 또한 레이더 운용에 필요한 각종 타이밍신호 및 모의 표적신호를 발생시킬 수 있는 TSG(timing signal generator)를 이용하여 도플러 처리기의 실시간 연산기능을 확인하였다.

### Abstract

Doppler processing is the heart of pulsed Doppler radar. It gives a clutter elimination and coherent integration. With the improvement of digital signal processors (DPSs), the implementation using them is more widely used in radar systems. Generally, so as for Doppler processor to process the input data in real time, a parallel processing concept using multiple DSPs should be used. This paper implements a programmable Doppler processor, which consists of MTI filter, DFB and square-law detector, using 8 ADSP21060s. Formulating the distribution time of the input data, the transfer time of the output data and the time required to compute each algorithm, it estimates total processing time and the number of required DSP. Finally, using the TSG that provides radar control pulses and simulated target signals, performances of the implemented Doppler processor are evaluated.

**Keyword** : Radar, Doppler processor, MTI filter, Digital signal processor

\* 正會員, 國防科學研究所

(Agency for Defense Development)

\*\* 正會員, 忠南大學校 情報通信工學部

(Division of Electrical and Computer Engineering,  
Chungnam National University)

接受日字:2003年5月27日, 수정완료일:2003年8月27日

I. 서론

도플러 필터링과 펄스압축 기능은 레이더 신호처리기에 적용되고 있는 최신 신호처리 기술로 분류될 수 있다. 레이더 시스템이 이들 두 가지 기법을 모두 사용하기 위해서는, 파형에 대한 위상정합이 펄스구간 내(intra-pulse)에서 뿐만 아니라 펄스간(inter-pulse)에도 유지되어야 한다. 이러한 두 기법은 잡음신호에 비해 큰 처리이득을 나타내기 때문에, 레이더의 성능향상 뿐만 아니라 송신 파형에 정합되지 못한 재밍신호에 대해서는 대 전자전(ECCM : electronic counter-countermeasures) 기능을 제공한다<sup>[1]</sup>.

펄스 도플러 레이더는 CPI 동안 일정한 펄스주기로 위상정합 관계를 갖는 여러 펄스를 송신한 후, 이들 펄스에 대한 반향(echo)신호 전체를 기본 블럭(block)단위로 하여 위상정합 누적 기능을 수행한다. 도플러 처리기에 의한 누적기능은 처리이득 뿐만 아니라, 클러터 신호에 묻혀 수신되는 작은 표적 탐지를 위해 필수적인 속도구별 능력을 제공한다<sup>[2]</sup>. 이러한 펄스 도플러 처리기와 펄스압축기 등의 레이더 신호처리기는 DSP의 성능향상으로 이를 이용한 프로그램 가능한 구조로 변화해가고 있다<sup>[3]</sup>.

본 논문에서는, 다중 DSP를 탑재한 상용보드를 사용하여 MTI 필터와 DFB 기능을 모두 갖춘 프로그램 가능한 도플러 처리기를 구성하고 그 성능을 분석/시험하였다. DSP보드로는 Spectrum Signal Processing사의 Morocco-2 보드를 사용하였다. 이 보드는 아날로그 디바이스사의 ADSP-21060 8개를 탑재하고 있기 때문에, 병렬처리를 쉽게 구현할 수 있도록 하여준다. CPI 동안 수신된 입력데이터의 분배시간, 출력데이터의 전송시간을 분석하고 MTI 필터, DFB 및 제곱검출기 알고리즘 수행에 필요한 연산시간을 수식으로 표현하여, 전체 처리시간과 하드웨어 구현에 필요한 DSP 수를 예측하였다. 또한 모의표적 신호와 운용 타이밍신호를 발생시킬 수 있는 TSG를 사용하여 실시간 연산기능을 확인하였다.

II. 도플러 처리기

도플러 처리기에서의 위상정합 누적은 펄스 도플러 레이더에 있어서 가장 핵심적인 신호처리 기능에 해당

된다. MTI 필터와 DFB가 동시에 사용되는 경우, 적응 MTI 필터가 강한 지면 클러터를 제거하기 위해 사용되며, DFB는 위상정합누적 수행으로 잔류 클러터를 제거하여 출력에서의 신호 대 클러터 비를 향상시킨다<sup>[4]</sup>. MTI 필터로는 2-펄스 또는 3-펄스 제거기(canceller)가 널리 사용된다. 2-펄스 제거기의 출력  $y(t)$ 는

$$y(t) = x(t) - x(t - T) \tag{1}$$

과 같이 표현되며, 식에서  $x(t)$ 는 제거기의 입력신호,  $T$ 는 펄스 간 시간간격인 PRI(pulse repetition interval)를 각각 의미한다. <그림 1>은 식 (1)의 관계를 갖는 2-펄스 제거기의 주파수응답 특성으로서 도플러 성분이 영(zero)에 가까운 강한 지면 클러터를 효과적으로 제거할 수 있다<sup>[5]</sup>.

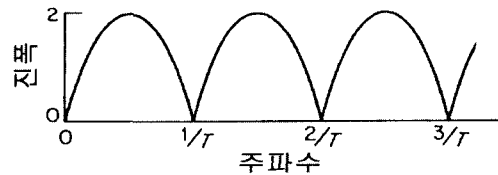


그림 1. 2-펄스 제거기의 주파수 응답특성  
Fig. 1. Frequency response of a 2-pulse canceller.

DFB가 영에서 펄스반복 주파수(PRF : pulse repetition frequency)  $f_p$  (또는  $1/T$ )까지의 주파수영역을  $N$ 개의 인접한 필터로 처리하기 위해서, <그림 2>의 구성과 같이  $N$ 개의 출력을 갖는 트랜스버설(transversal) 필터가 사용될 수 있다.  $N$ -탭을 갖는 트랜스버설 필터를 사용하여  $N$ 개의 디지털 필터를 형성하기 위해서는,  $(N-1)^2$  번의 곱셈이 필요하며, 이는 DFT(discrete Fourier transform) 연산과 동일하다. 따라서  $N$ 이 2의 멱승(power)인 경우, FFT(fast Fourier

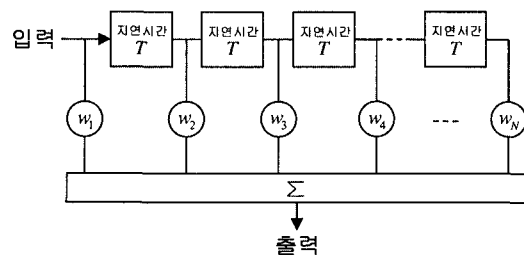


그림 2. 트랜스버설 필터의 일반적 구성  
Fig. 2. General form of a transversal filter.

transform) 알고리즘이 사용될 수 있으며, 요구되는 곱셈 수는  $(N-1)^2$ 에서  $N/2 \cdot \log_2 N$  으로 줄어들게 된다<sup>[6]</sup>.

<그림 3>은 구성하고자 하는 도플러 처리기의 기능별 구성도를 나타내고 있다. ADC(analog-to-digital converter) 또는 펄스압축기로부터 전달되는 입력 데이터는  $N_R \cdot N_p$  복소수 데이터(I 및 Q)로 구성되며,  $N_R$ 은 각 펄스에서 수신되는 거리 셀 수를,  $N_p$ 는 CPI 동안 수신되는 펄스 수를 각각 의미한다. 도플러 처리기는  $N_R \cdot N_p$ 개의 데이터를 수신한 후 이를 기본단위로 처리하며, 일반적으로 실시간 수행을 위해 다중 DSP를 사용하여 구현된다. 즉,  $N_p$  펄스에 대한 누적연산을 수행하기 위해서는 입력데이터를 <그림 4>에서와 같이 거리 셀 방향에서 분할하여 여러 DSP로 나누어주어야 한다. 그러나 ADC 또는 펄스압축기로부터의 데이터 전달은 각 펄스에 대해 거리 셀 방향으로  $N_R$ 개씩 이루어진다. 따라서 입력되는 데이터를 순차적으로 메모리에 저장하는 경우, 각 DSP로 전송될 데이터가 메모리 상에 연속하여 위치하지 못하게 된다. 이는 연산 DSP 들로의 블록단위 데이터 전송을 어렵게 하므로 전송속도의 저하를 초래하게 된다. 따라서 입력데이터를 동일 거리 셀의  $N_p$ 개 펄스 순으로 재 배열시켜 주는 "corner turn" 수행이 여러 DSP 들로의 데이터 전송을 매우 용이하게 하여준다<sup>[7]</sup>.

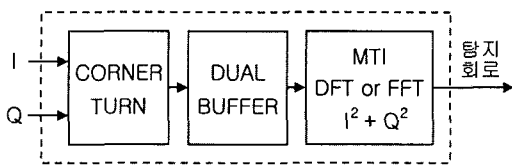


그림 3. 도플러 처리기의 기능별 블록도  
Fig. 3. Functional block diagram of the Doppler processor.

도플러 처리기 구현을 위해  $N$  개의 DSP를 사용하는 경우, 이에 대한 타이밍 다이어그램은 <그림 5>와 같이 일반화될 수 있다. <그림 5>에서와 같이 입력 데이터를  $N$  등분한 블록데이터가 연산 DSP로 분배되며, 데이터 수신을 완료한 DSP부터 연산을 시작하여 연산 완료 순서에 따라 출력 데이터를 다음 처리기로 전송한다. 이 경우 실시간 연산이 이루어지기 위해서는 <그림 5>에서의 전체 처리시간이 CPI 보다 작아야 함을 알 수 있다.

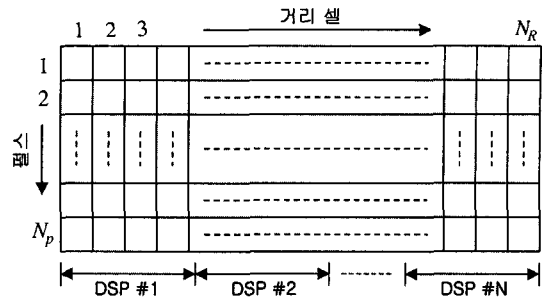


그림 4. 입력데이터의 분할  
Fig. 4. Partition of the input data.

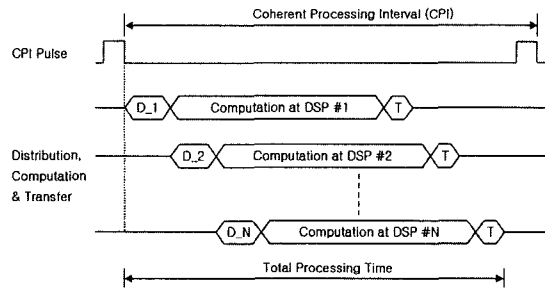


그림 5. 도플러 처리기의 타이밍도  
Fig. 5. General timing diagram for a Doppler processor.

### III. 도플러 처리기 구성

<그림 3>의 기능을 갖는 프로그램 가능한 도플러 처리기를 구성하기 위하여, Spectrum Signal Processing사의 DSP 보드인 Morocco-2를 사용하였다. 프로그램 개발 및 디버깅은 윈도우 NT 환경 하에서 ADI VisualDSP 4.1 과 APEX Pro V3.2를 사용하였다.

Morocco-2는 아날로그 디바이스사의 ADSP- 21060 8개를 탑재한 VME 버스 기반의 DSP 보드이다. 특히 보드 상에 PMC(PCI mezzanine card) 모듈을 탑재할 수 있도록 되어 있기 때문에, 하나의 슬롯에서 ADC 등의 I/O 문제를 해결할 수 있다. 이 보드에 대한 주요 특징은 다음과 같다<sup>[8]</sup>.

- (1) 4개의 클러스터(cluster)로 구성되어 있으며, 각 클러스터에는 2개의 ADSP21060이 위치하여 총 8개의 40MHz ADSP-21060이 탑재,
- (2) 최대 연산능력 960MFLOPS, 실용 연산능력 640MFLOPS,
- (3) 클러스터 당 512KB, 1MB 또는 2MB SRAM (클러스터 내의 DSP가 공유),

- (4) PMC, VME64 및 클러스터에 위치한 ADSP-21060의 SRAM 접근 가능
- (5) 링크포트를 이용한 프로세서간 데이터 전달 (40Mbytes)
- (6) 보드 상에 PMC 모듈 탑재 가능.

<그림 6>은 구성된 도플러 처리기의 링크(link)구성 및 데이터 흐름을 보여주고 있다. <그림 6>에서 클러스터 1의 DSP 1 또는 PN(processor number): 19가 펄스압축기로부터 전송된 데이터를 수신하며, 이들 데이터는 CPI 단위로 번갈아 가며 클러스터 2의 DSP 0 (PN: 22) 과 DSP 1 (PN: 20)로 전달된다. <그림 3>에서의 이중버퍼(dual buffer) 기능은 이와 같은 데이터 전달로 구현되었다. 특히, 이 경우 PN: 19가 각 펄스에 대한 거리 셀 데이터를  $N$  등분한 분할 데이터를 PN: 20 또는 PN: 22로 전송하므로, "corner turn"과 동일한 효과가 발생되도록 하였다.  $N$ 은 도플러 처리기의 연산 수행에 사용되는 DSP의 개수이며, 본 구성에서는 6이 된다. <그림 7>은 6개의 DSP를 사용한 경우 분할 데이터 전송에 의한 데이터의 재 배열과정을 보여주고 있다. 6개의 DSP로 분배될 데이터가 PN: 20 또는 PN: 22에서 연속성을 가지면서 저장되기 때문에 데이터의 블럭전송이 쉽게 이루어질 수 있다.

<그림 6>에서 PN: 20 또는 PN: 22가  $N_p$  펄스에 대한 데이터 수신을 완료하면, 해당 프로세서는 수신된 데이터의 5/6를 나머지 5개, PN: 16, PN: 17, PN: 18, PN: 21 및 PN: 23, DSP로 분배하며 데이터 수신을 완료한 DSP는 연산을 시작하게 된다. PN: 20으로부터 PN: 16 및 PN: 22로부터 PN: 18로의 데이터 전송은 <그림 6>에서처럼 PN: 17을 경유하여 이루어진다. 연산결과를 다음 처리기인 일정 오경보율 (CFAR: constant false alarm rate) 처리기로 전송하는 역할은 짝수 CPI에 대해서는 PN: 21이, 홀수 CPI에 대해서는 PN: 23이 각각 담당하도록 하였다.

도플러 처리기 운용을 위한 시스템 프로그램은 C-언어와 어셈블리어언어를 혼용하여 개발되었다. C-언어는 프로그램 구성 등에 사용되었으며, 어셈블리 언어는 32-비트 IEEE 부동소수점 연산 및 데이터 전송을 위해 사용되었다. DSP 간의 데이터 전달은 APEX에서 제공하는 RAWLINK 함수 및 다중 프로세서 메모리 공간 개념(multi-processor memory space concept)을 사용하여 구현되었다.

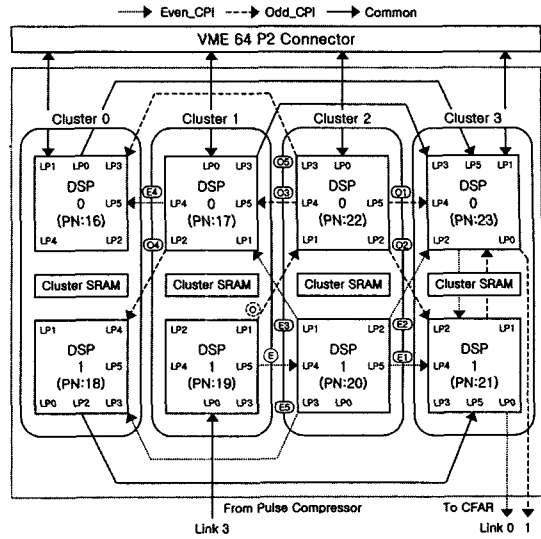


그림 6. 도플러 처리기의 링크 구성 및 데이터 흐름  
Fig. 6. Link configuration and data flow of the implemented Doppler processor.

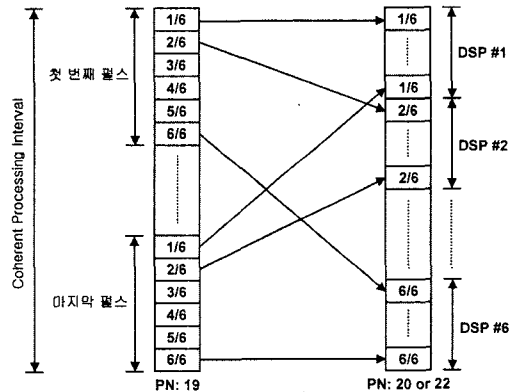


그림 7. 입력 데이터의 재배열  
Fig. 7. Rearrangement of the input data.

#### IV. 도플러 처리기의 연산시간 분석

도플러 처리기에 대한 연산능력을 분석하기 위해서는 구성된 하드웨어의 타이밍 분석과 연산시간에 대한 수식화가 필요하다. 본 논문에서는 강한 지면 클러터 제거를 위한 MTI 필터로는 식 (1)의 2-펄스 제거기를 사용하였으며, CPI 내에서 수신되는 펄스 수의 제약을 없애기 위하여, DFT를 도플러 필터뱅크 구현에 사용하였다. 한 개의 ADSP21060을 사용하여, 식 (1)의 2-펄스 제거 알고리즘을 수행하기 위해 소요되는 연산시간  $T_{min}$ 는

$$T_{mi} = \frac{2 N_R \{3 (N_P - 1) + 5\} + 15}{40} [\mu s] \quad (2)$$

과 같으며, DFB 연산과  $I^2 + Q^2$  연산이 필요한 제곱검출기 수행에 필요한 시간  $T_{dfb}$  는

$$T_{dfb} = \frac{N_R \{ (N_P - 1) \{4(N_P - 2) + 13\} + 3\} + 29}{40} [\mu s] \quad (3)$$

과 같다. 이들 수식은 ADI VisualDSP 4.1 환경에서 개발된 어셈블리 언어의 반복 루프를 기초로 하여 얻은 결과이며, 덧셈 항으로 추가되어 있는 상수 값들은 초기화 과정에서 필요한 클럭(clock) 수들로서 디버거를 이용하여 얻어진 값들이다.

짝수 CPI에 대한 도플러 처리기의 타이밍 구성은 <그림 8>과 같다. 즉, 수신된 데이터의 5/6는 <그림 6>의 PN:20으로부터 나머지 5개 DSP로 분배되며, 그 순서는 <그림 6>과 <그림 8>에 나타나 있으며, 경로 E4와 E5에서의 데이터 전송은 동시에 이루어진다. APEX의 RAWLINK 함수가 DMA(direct memory access) 컨트롤러를 사용하기 때문에, <그림 6>의 PN:17 및 PN:20에서의 연산은 데이터 전송과 동시에 시작됨을 알 수 있다. 링크 포트를 통한 데이터 전송률이 40Mbps (mega bytes per second)임으로, 입력 데이터를 분배하는데 소요되는 시간  $T_D$ 는 <그림 8>로부터

$$T_D = \frac{5}{6} \frac{8 (N_R N_P)}{40} [\mu s] \quad (4)$$

과 같이 주어진다. 식에서 분자에 나타난 상수 8은 두 채널(I 및 Q)데이터가 32-비트 부동소수점으로 표현되기 때문에 나타난 결과이다.

짝수 CPI인 경우, <그림 6>에서 클러스터 3의 DSP 1 (PN:21)이 출력 데이터를 CFAR 처리기로 전달하는 역할을 담당하며, CFAR 처리기로 전달되는 순서는 <그림 8>에 나타나 있다. <그림 6>에서 PN:16과 PN:17의 출력 데이터는 PN:23을 경유하여 PN:21로 전달되며, PN:17에서 PN:23, PN:20에서 CFAR 처리기로의 데이터 전송은 동시에 이루어짐을 알 수 있다. 또한 PN:16에서 PN:23, PN:18에서 CFAR 처리기로의 데이터 전송도 동시에 수행된다. 따라서 모든 DSP가 연산을 끝낸 이후, 출력 데이터를 CFAR 처리기로 보내는데 필요한 전송시간  $T_{ET}$ 는 다음과 같이 표현될 수 있다.

$$T_{ET} = \left\{ 2 \left( \frac{1}{80} + \frac{1}{40} \right) + \frac{2}{40} \right\} \frac{4 N_R (N_P - 1)}{6} [\mu s] \quad (5)$$

식 (5) 에서, 1/80은 다중 프로세서 메모리 개념에 의한 데이터 전송률을 1/40은 링크포트의 데이터 전송률을 각각 나타내며, 상수 2는 PN:16 및 PN:17 두 DSP로부터의 출력 전송에 의한 결과이다. 2/40은 PN:18의 출력 데이터가 PN:21을 경유하여 CFAR 처리기로 전달되는 경우의 전송률을 각각 의미하며, 마지막 항에서의 분자 4는 제곱 검출기에 의해 복소수 데이터가 실수 데이터로 변환됨에 따라 식 (4)의 8이 반으로 줄어든 결과이다. 그리고 1/6은 각 DSP의 데이터 양이 전체의 1/6만을 처리하기 때문에 나타난 결과이며,  $(N_P - 1)$ 은 2-펄스 제거기에 의해 유효 펄스수가 하나 줄어들었기 때문에 나타난 결과이다. 따라서 식 (2), 식 (3), 식 (4) 및 식 (5)를 사용하여 짝수 CPI인 경우의 전체 처리시간을 예측할 수 있다.

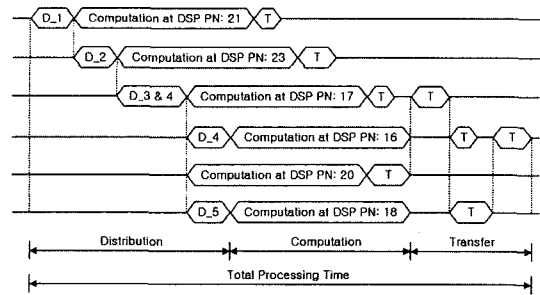


그림 8. 짝수 CPI에 대한 도플러 처리기의 타이밍 다이어그램

Fig. 8. Timing diagram of the implemented Doppler processor for even CPIs.

홀수 CPI인 경우, <그림 6>에서 PN:20 과 PN:23 이 입력 데이터 및 출력 데이터의 전송을 각각 담당하고 있으며, 타이밍 구성은 <그림 9>와 같다. 데이터 분배 및 연산시간은 짝수 CPI와 동일하며, PN:18의 출력 데이터는 PN:21을 경유하여 PN:23으로 전달된다. 따라서 모든 DSP가 연산을 끝낸 이후, 출력 데이터를 CFAR 처리기로 보내는데 필요한 전송시간  $T_{OT}$ 는 다음과 같이 표현될 수 있다.

$$T_{OT} = \left\{ \left( \frac{1}{80} + \frac{1}{40} \right) + 2 \frac{2}{40} \right\} \frac{4 N_R (N_P - 1)}{6} [\mu s] \quad (6)$$

결국, 식 (2), 식 (3), 식 (4) 및 식 (6)을 사용하여 홀

수 CPI에 대한 전체 처리시간을 예측할 수 있다.

식 (5) 및 식 (6)으로부터 홀수 CPI인 경우의 처리시간이 크게 나타남으로 실시간 처리를 위해서는

$$\frac{T_{mti} + T_{dfb}}{CPI - T_D - T_{OT}} \leq 6 \quad (7)$$

의 관계가 만족되어야 함으로, 실시간 처리여부를 쉽게 확인할 수 있다.

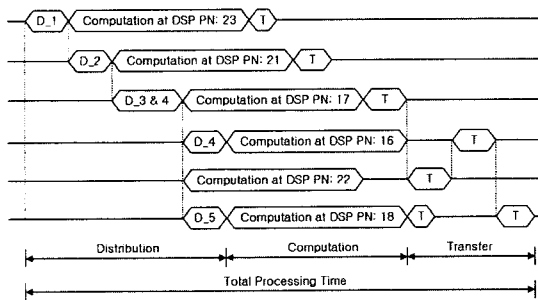


그림 9. 홀수 CPI에 대한 도플러 처리기의 타이밍 다이어그램

Fig. 9. Timing diagram of the implemented Doppler processor for odd CPIs.

V. 시스템 적용 및 성능확인

중거리 탐색(surveillance)레이더에 사용될 수 있는 <표 1>의 파라미터를 이용하여, 본 논문에서 구성한 하드웨어의 시스템 적용 가능성을 확인하였다. 식 (4), 식 (5) 및 식 (6)을 사용하여, 입력데이터를 분배하고 출력 데이터를 전송하는데 소요되는 시간을 계산할 수 있으며, 6개의 ADSP21060이 알고리즘을 병렬로 처리하기 때문에, 연산시간은 식 (2) + 식 (3)의 1/6이 된다. 결국, 전체 처리시간이 짝수 CPI 인 경우에는 4.4ms, 홀수 CPI인 경우에는 4.484ms 가 된다. 이 시간은 6ms인 CPI 보다 작음을 알 수 있으며, 식 (7)을 만족하는 최소 DSP는 3.05가 되어 실시간 수행을 위해 4 개 이상의 DSP가 소요됨을 알 수 있다. 따라서 구성된 도플러 처리기는 <표 1>의 파라미터를 갖는 탐색 레이더의 MTI 필터, DFB 및 제곱검출기 알고리즘을 실시간으로 수행할 수 있음을 알 수 있다.

개발된 도플러 처리기의 실시간 연산 수행기능을 확인하기 위하여, 기계적으로 회전하는 안테나를 사용하는 펄스 누적 레이더의 각종 타이밍 신호를 발생할 수 있는 TSG를 사용하였다. 제작된 TSG의 구성은 1) 방

표 1. 도플러 처리기와 관련된 레이더 파라미터

Table 1. Radar parameters related to Doppler processor.

항목	설정값
안테나 회전속도	12 RPM
안테나 방위각 빔폭	1.5 도
CPI 당 펄스 수	6
펄스반복 주파수	1,000 Hz
ADC 샘플링 주파수	2.5 Mz
처리되는 거리 셀 수	2000 개 (120 Km)

위각 기준펄스(ARP: azimuth reference pulse) 및 방위각 변화 펄스(ACP: azimuth changing pulse) 발생부, 2) 드웰 시간(DT: dwell time) 펄스 발생부, 3) 송신펄스 발생부, 4) 위상정합처리구간(CPI) 펄스 발생부, 5) 클럭 발생부, 6) 표적 트리거펄스 발생부, 7) 모의 표적 신호 발생부, 8) VME 인터페이스 등 8개 블록으로 구성되어 있으며, 각종 타이밍 신호들은 프로그램에 의해 가변 될 수 있도록 설계/제작하였다<sup>[9]</sup>.

도플러 처리기의 기능을 실시간으로 시험하기 위하여 <표 1>의 파라미터가 발생되도록 TSG를 설정한 후, TSG에서 발생된 모의 표적신호가 ADC모듈로부터 펄스압축 처리기를 지나 도플러 처리기로 인가되도록 구성하였다. <그림 10>은 이러한 환경 하에서 홀수 CPI에 대해 식 (2)~식 (6)의 관계식을 이용하여 계산한 전체 처리시간과 APEX Pro에서 측정된 실제 하드웨어 처리시간을 비교한 결과로서 두 결과가 일치함을 확인할 수 있다.

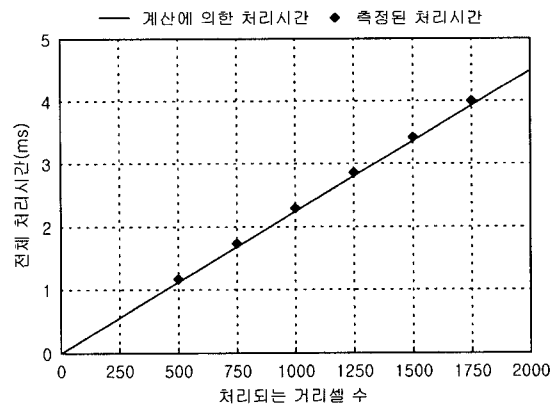


그림 10. 도플러 처리기의 전체 처리시간.

Fig. 10. Total processing time of Doppler processor.

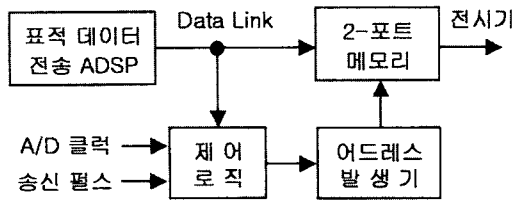


그림 11. 인터페이스 보드의 구성  
Fig. 11. Block diagram of the interface board.

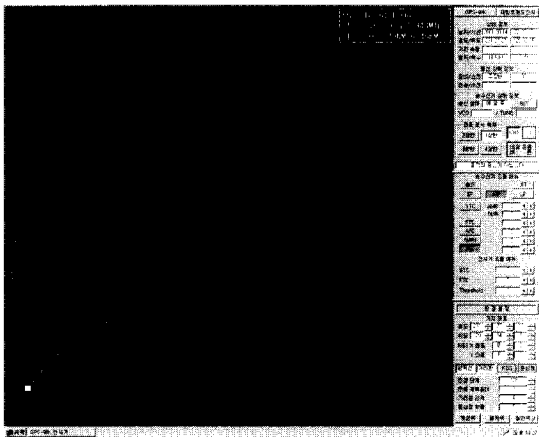


그림 12. 구현된 도플러 처리기에 의한 표적탐지  
Fig. 12. Target detection using the implemented Doppler processor.

또한 <그림 11>의 구성을 갖는 인터페이스 보드를 제작하여, CFAR 처리기의 ADSP21060으로부터 전송된 표적 탐지데이터를 저장하고, 이를 TSG의 운용 타이밍 신호(A/D 클럭 및 송신펄스)에 동기시켜 탐지표적이 전시기 상의 정확한 거리에 나타날 수 있도록 하였다. 그리고 모의표적신호의 거리 셀 및 CPI 발생위치가 매 스캔(scan) 마다 하나씩 감소하도록 TSG의 파라미터를 프로그램 하여 도플러 처리기의 실시간 시험을 수행하였다. <그림 12>는 이러한 설정 하에서 탐지된 표적신호의 궤적을 전시결과로서 도플러 처리기의 실시간 연산수행 결과를 확인할 수 있었다.

### VI. 결 론

ADSP21060을 이용한 프로그램 가능한 도플러 처리기에 대한 구성, 분석 및 성능시험을 수행하였다. ADSP21060 8개를 탑재한 다중 DSP 보드를 사용함으로써, 병렬처리 구성을 쉽게 구현할 수 있었다. 특히 분배 DSP로의 데이터 전달 시 데이터를 등분하여 전송

함으로써, "corner turn"과 동일한 효과가 나타나도록 하여 DSP들로의 데이터 전송시간을 최소화하였다. CPI 동안 수신된 입력데이터의 분배시간, 출력데이터의 전송시간 및 알고리즘 수행에 소요되는 연산시간을 최소화 하여, 전체 처리시간 및 소요 DSP 수를 쉽게 예측하였다. 또한 TSG를 사용하여 처리시간에 대한 예측결과를 검증하였으며, 전시기에서 최종표적 탐지결과를 실시간으로 확인하여 구성된 도플러 처리기가 탐색레이더 시스템에 적용될 수 있는 가능성을 확인하였다. 상용의 DSP 보드를 사용하였기 때문에, 하드웨어 측면에서는 최적의 구성이라 할 수 없다. 하지만, 프로그램 가능한 특징을 가지고 있기 때문에 펄스 도플러 레이더의 성능확인용 에뮬레이터(emulator) 또는 최신 레이더에 대한 재밍효과 시험분야에 효과적으로 활용될 수 있을 것으로 판단된다.

### 참 고 문 헌

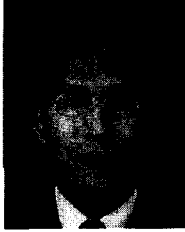
- [1] D.C. Schleher, Electronic Warfare in the Information Age, Atrech House, Norwood, MA, 1999.
- [2] P.E. Rademacher, B.H. Cantrell, and G.C. Tavik, "Cluttering filtering and Processing Techniques for EMI Detection and Angle Measurement in Pulse Doppler Radars," IEEE National Radar Conference, pp. 273-278, 1996.
- [3] H.I. Shin et. al., "Comparison of Computation Capability of Expansible Digital Compressors Using ADSP21060," Workshop on DSP in Mobile and Vehicular Systems, CD, Nagoya, Japan, April 2003.
- [4] Y.K. Kwag, T.I. Uam, H.I. Shin, and S.C. Park, "A Programmable Pulsed Doppler Radar Signal Processor Implementation with DSP for Adaptability and Multi-Function," IEE Radar Conference, pp. 379-382, 1992.
- [5] N. Levanon, Radar Principles, John Wiley & Sons, Inc., 1988.
- [6] M.I. Skolnik, Introduction to Radar Systems, McGraw-Hill, Inc., New York, NY, 1980.
- [7] C. Buenzli, L. Owen, and F. Rose, "Hardware/Software Codesign of a Scalable Em-

- bedded Radar Signal processor," VHDL international Users Forum, pp. 200-208, 1997.
- [8] Analog Device Inc., ADSP-2106X Users Manual, ADI, 2nd Edition, 1997.
- [9] 신현익, 임중수, 김환우, "위상정합 레이더에 대한 EA효과 분석용 타이밍 신호발생기," 전자공학회 논문지, 제39권 SC편, 제2호, 32- 38쪽, 2002년 3월

---

 저 자 소 개
 

---



申鉉翼(正會員)

1986년 : 경북대학교 전자공학과 학사. 1988년 : 경북대학교 전자공학과 석사. 1999년~현재 : 충남대학교 대학원 전자공학과 박사과정. 1988년~현재 : 국방과학연구소 선임연구원. <주관심분야 : 레이더 시

스템 설계/분석, 신호처리 및 시험평가 기법개발, 레이더 재밍장치 설계>



金煥宇(正會員)

1977년 : 서울대학교 전자공학과 학사. 1979년 : 한국과학기술원 전기 및 전자공학과 석사. 1988년 6월 : University of Utah 전자공학과 공학박사. 1980년 6월~현재 : 충남대학교 정보통신공학부 교수.

1998년~현재 : 한국통신학회 대전충남지부 이사. 2000년~현재 : 대한전자공학회 충남지부 이사. 2002년~현재 : IEEE 대전섹션 부의장. <주관심분야 : 디지털 신호처리, 디지털 통신 >